International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

A Comprehensive Study of Statistical Machine Translation for English to Kannada Language

Chitra C[1], Shiva Kumar K M[2]

MCA Student, Amrita School of Arts and Sciences

Faculty of PG, PhD Scholar, Amrita Vishwa Vidya Peetham, Mysore Campus

Abstract: ***The field of machine translation has recently been energized by the emergence of statistical techniques. Development of a Machine Translation (MT) system for any two natural languages is a challenging task. This paper presents the impact of Mathematics (Statistics) on Statistical Machine Translation for English to Kannada Language. MT has its own approach to process the Corpus data for translation, whereas the whole approach of SMT is built on Statistical rules rather than linguistically motivated rules. It provides the necessary grounding in linguistics and probabilities.***

Keywords: SMT, Translation Modeling, Evaluating, Language Modeling, Decoding, BLEU

1. Introduction

Statistical Machine Translation is a technique that uses parallel corpora (documents in one language paired with their translations into another language) to automatically induce bilingual dictionaries and translation rules. By analyzing the co-occurrence and relative orderings of words in large amounts of such texts a statistical model of the translation process can be approximated. These approximations are done through Probabilistic measures in each module of SMT process.

A. Basic Probability

We're going to consider that an English sentence 'e' may translate into any Kannada sentence 'k'. Some translations are just more likely than others. Here are the basic notations we'll use to formalize "more likely":

1) P(e) - a priori probability: The chance that e happens. For example, if e is the English string "I like books," then P(e) is the chance that a certain person at a certain time will say "They do it" as opposed to saying something else[7].

2) P(k|e) - conditional probability: The chance of 'k' given 'e'. For example, if 'e' is the English string "They do it" and if 'k' is the Kannada string "ಅವರು ಇದನ್ನು ಮಾಡುತ್ತಾರೆ" then P(k|e) is the chance that upon seeing 'e', a translator will produce 'k'[7].

3) P(e,k) -- joint probability: The chance of 'e' and 'k' both happening. If 'e' and 'k' don't influence each other, then we can write P(e,k) = P(e) * P(k). For example, if e stands for "the first roll of the die comes up 5" and k stands for "the second roll of the die comes up 3," then P(e,k) = P(e) * P(k) = 1/6 * 1/6 = 1/36. If e and k do influence each other, then we had better write P(e,k) = P(e) * P(k | e). That means: the chance that "e happens" times the chance that "if e happens, then k happens." If e and k are strings that are mutual translations, then there's definitely some influence [7].

All these probabilities are between zero and one, inclusive. A probability of 0.5 means "there's a half a chance."

B. Sums and Products

1) To represent the addition of integers from 1 to n, we write:

$$\sum_{i=1}^{n} i = 1 + 2 + 3 + \ldots + n$$

2) For the product of integers from 1 to n, we write:

$$\prod_{i=1}^{n} i = 1 * 2 * 3 * \ldots * n$$

3) If there's a factor inside a summation that does not depend on what's being summed over, it can be taken outside:

$$\sum_{i=1}^{n} i * k = k + 2k + 3k + \ldots + nk = k \sum_{i=1}^{n} i$$

2. Statistical Machine Translation

Given an English sentence 'e', we seek the Kannada sentence 'k' that maximizes P(k|e). (The "most likely" translation). Sometimes we write:

$$\arg_k \max P(k \mid e)$$

Read this argmax as follows: "the Kannada sentence k, out of all such sentences, which yields the highest value for P(k|e). If you want to think of this in terms of computer programs, you could imagine one program that takes a pair of sentences 'k' and 'e', and returns a probability P(k|e).

3. Language Modeling

First we need to build a machine that assigns a probability P(k) to each Kannada sentence 'k'. This is called a Language Model. The probability is computed using n-gram model. Language Model can be considered as computation of the probability of single word given all of the words that precede it in a sentence. The goal of Statistical Machine Translation is to estimate the probability (likelihood) of a sentence. A sentence is decomposed into the product of conditional probability. By using chain rule, this is made

possible as shown below. The probability of sentence P (S), is broken down as the probability of individual words P(w)[2].

P(s) = P(w1, w2, w3,....., wn)
=P(w1) P(w2|w1) P(w3,|w1w2) … P(wn|w1w2…wn-1))

In order to calculate sentence probability, it is required to calculate the probability of a word, given the sequence of word preceding it. An n-gram model simplifies the task by approximating the probability of a word given all the previous words. An n-gram of size 1 is referred to as a unigram; size 2 is a bigram (or, less commonly, a digram); size 3 is a trigram; size 4 is a four-gram and size 5 or more is simply called a n-gram

Consider the following training set of data given

ಇದು ಕೆಂಪು ಗುಲಾಬಿ

ಇದು ಸುಂದರವಾದ ಕೆಂಪು ಗುಲಾಬಿ

ಇದು ಸುಂದರವಾದ ಹೂವು

A. Parameters identified by unigram model are as shown below:

(ಇದು), (ಕೆಂಪು), (ಗುಲಾಬಿ), (ಸುಂದರವಾದ), (ಹೂವು)

B. Parameters are identified by bigram model are as shown below:

(ಇದು | <Start of sentence>), (ಕೆಂಪು|ಇದು), (ಸುಂದರವಾದ | ಇದು), (ಗುಲಾಬಿ | ಕೆಂಪು), (ಇದು | ಗುಲಾಬಿ), (ಕೆಂಪು | ಸುಂದರವಾದ), (ಹೂವು | ಸುಂದರವಾದ), (<End of sentence> | ಹೂವು), (<End of sentence> | ಗುಲಾಬಿ)

C. Parameters identified by trigram model are as shown below:

(ಗುಲಾಬಿ | ಇದು ಕೆಂಪು)

(ಗುಲಾಬಿ | ಇದು ಸುಂದರವಾದ)

For all these parameters the probabilities are calculated by the Language Model toolkits like IRSTLM, SRILM etc. The probability of a sentence: 'ಇದು ಸುಂದರವಾದ ಗುಲಾಬಿ', can be computed as follows, using Bigram computation:

= P(ಇದು | <start of sentence>) * P(ಸುಂದರವಾದ | ಇದು) * P(ಗುಲಾಬಿ | ಸುಂದರವಾದ) * P(<End of sentence> | ಗುಲಾಬಿ)

4. Translation Modeling

Finding P (e | k), the probability of an English string 'e' given a Kannada string k. This is called Translation model. Translation Model is trained in following steps:

1) Word to Word alignment
2) Phrase pair extraction
3) Scoring Phrase Translation

The Translation Model helps to compute the conditional probability P(e|k). It is trained from parallel corpus of target-source pairs. As no corpus is large enough to allow the computation translation model probabilities at sentence level, so the process is broken down into smaller units, e.g., words or phrases and their probabilities learnt. The target translation of source sentence is thought of as being generated from source word by word[2]. For example, using the notation (T/S) to represent an input sentence S and its translation T. Using this notation, sentence is translated as given below.

(ಇದು ಕೆಂಪು ಗುಲಾಬಿ | this is red rose)

One possible alignment for the pair of sentences can be represented as given below

(ಇದು ಕೆಂಪು ಗುಲಾಬಿ | this(1) is(1) red(2) rose(3))

A number of alignments are possible. For simplicity, word by word alignment of translation model is considered. The above set of alignment is denoted as A(S, T). If length of target is 'l' and that of source is 'm' than there are 'l*m' different alignments are possible and all connection for each target position are equally likely, therefore order of words in T and S does not affect P (T|S) and likelihood of (T|S) can be defined in terms of the conditional probability P (T, a/S) as shown in below

$$P(S|T) = \sum P(S, a/T)$$

The sum is over the elements of alignment set, A(S, T). English word has only exactly one connection. For the alignment, (ಇದು ಕೆಂಪು ಗುಲಾಬಿ | this is red rose), can be computed by multiplying the translation probabilities P(ಇದು | this(1)), P(null | is(1)), P(ಕೆಂಪು | red(2)), P(ಗುಲಾಬಿ|rose(3))

To generate target sentence from source sentence, we have to follow the steps as given below:

1) Select the length of S with probability L where L=P[length(S) = m] is a constant i.e. All lengths are assumed to be equally likely with probability L.
2) Select an alignment with probability P(a|S). There are (l+1)m possible alignments. Assuming all possible alignments are equally likely, the probability of alignment a, P(a|S), is as shown below
$$P(a|S) = L \times 1/(l+1)m$$
3) Select the jth English word with a probability. The joint likelihood of Kannada string and an alignment given an English string is given below
$$P(S, a/T) = P(a/T)*P(S/a, T)$$
T is the probability of seeing Sj in source sentence, given Taj in target sentence. The alignment is determined by specifying the values of aj for j from 1 to m, each of which can take value from 0 to l.

5. Decoding

Job of decoder is to find the highest scoring sentence in the target language corresponding to source sentence. It uses the phrase translation table generated during the training of translation model. The system can learn parameters values for computing P(k) from monolingual Kannada text. Similarly it can learn parameter values for computing P(e | k) from bilingual sentence pairs. To translate an observed English sentence e, we seek the Kannada sentence k which maximizes the product of those two terms. This process is called decoding.

6. Evaluation

The evaluation for the Statistical Machine Translation tool for English to Kannada is done by BLEU-Bilingual Evaluation Understudy tool. BLEU's output is always a

number between 0 and 1. This value indicates how similar the candidate and reference texts are, with values closer to 1 representing more similar texts.

BLEU is really just the geometric mean of n-gram precisions that is scaled by a brevity penalty to prevent very short sentences with some matching material from being given inappropriately high scores. Since the geometric mean is calculated by multiplying together all the terms to be included in the mean, having a zero for any of the n-gram counts results in the entire score being zero.

7. Experimental Results

A) Language Model

As explained in the above section, Let the training set be
ಇದು ಕೆಂಪು ಗುಲಾಬಿ

ಇದು ಸುಂದರವಾದ ಕೆಂಪು ಗುಲಾಬಿ

ಇದು ಸುಂದರವಾದ ಹೂವು

Probabilities for parameters of Unigram model are
P(ಇದು)=0.13, P(ಕೆಂಪು)=0.1, P(ಸುಂದರವಾದ)=0.1, (ಗುಲಾಬಿ)=0.1,

P(ಹೂವು)=0.068

The probability of a sentence: 'ಇದು ಸುಂದರವಾದ ಗುಲಾಬಿ',

can be computed as follows, using Unigram computation:
P(ಇದು) * P(ಸುಂದರವಾದ) * P(ಗುಲಾಬಿ)
= 0.13 * 0.1 * 0.1
= 0.0013

B) Translation Model

The important part of Translation Model is generating Phrase table. It involves following stages

1) Word to Word alignment

Phrase-based translation models are acquired from a word-aligned parallel corpus by extracting all phrase-pairs that are consistent with the word alignment. Given the set of extracted phrase pairs with counts, various scoring functions are estimated, such as conditional phrase translation probabilities based on relative frequency estimation or lexical translation probabilities based on the words in the phrases. In Moses, the models for the translation steps are acquired in the same manner from a word-aligned parallel corpus. For the specified factors in the input and output, phrase mappings are extracted. The Word Mapping can be represented as follows

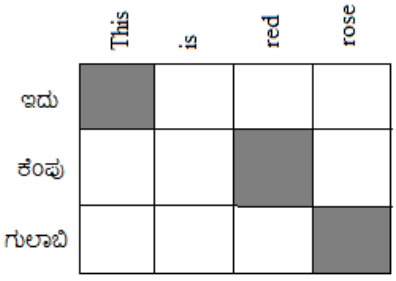

Figure 1: Word Mapping of English and Kannada Languages

2) Phrase Pair Extraction

The set of phrase mappings is scored based on relative counts and word-based translation probabilities. The phrase mappings are done by adding some additional alignment points that lie in the union of the two alignments. All words of the phrase pair have to align to each other. That is the phrase pair must be consistent, the inconsistent phrase pairs are violated.

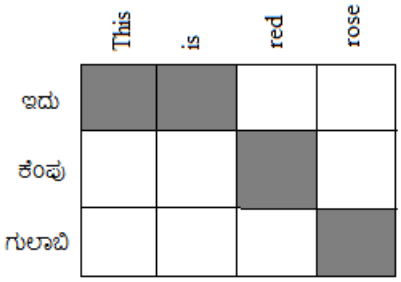

Figure 2: Phrasal pair Extraction

3) Scoring Phrase Translation

The phrase pairs identified in the previous stage have to be extracted. The probabilities have to be assigned for every phrase pair. This is done through scoring by relative frequency

$$\varphi(e^I|k^I) = \text{count}(k^I, e^I) / \sum e^I i \text{ count}(k^I, e^I i)$$

Generating the Phrase Translation table is the next step. This is done by dumping all the phrase pairs along with its scores in a big file called Phrase Translation Table. During this step a config file is created, which is very much helpful during the decoding process. For every phrase pairs a phrase entry is created in Phrase Translation table. The format of phrase entry is as shown below

source ||| target ||| scores ||| [alignment] ||| [counts]

The alignment and count fields may are may not be created during translation modeling. Because these fields are not mandatory, these fields are responsible for improving translation quality. The below figure is the screenshot of phrase translation table for the above 3 sentences

```
is beautiful red ||| ಸುಂದರವಾದ ಕೆಂಪು ||| 0.5 1 1 1 ||| 1-0 2-1 ||| 2 1 1 |||
is beautiful ||| ಸುಂದರವಾದ ||| 0.5 1 1 1 ||| 1-0 ||| 4 2 2 |||
is red rose ||| ಕೆಂಪು ಗುಲಾಬಿ ||| 0.333333 1 1 1 ||| 1-0 2-1 ||| 3 1 1 |||
is red ||| ಕೆಂಪು ||| 0.333333 1 1 1 ||| 1-0 ||| 3 1 1 |||
red rose ||| ಕೆಂಪು ಗುಲಾಬಿ ||| 0.666667 1 1 1 ||| 0-0 1-1 ||| 3 2 2 |||
red ||| ಕೆಂಪು ||| 0.666667 1 1 1 ||| 0-0 ||| 3 2 2 |||
rose ||| ಗುಲಾಬಿ ||| 1 1 1 1 ||| 0-0 ||| 2 2 2 |||
```

Figure 3: Phrase Translation Table

C) Decoding

The decoder uses some intelligent algorithms to traverse through the huge Phrase translation table and search for the appropriate phrases. It's impossible to search through all possible sentences, but it is possible to inspect a highly relevant subset of such sentences. The Decoder does just that, and produces the single sentence from the subset that it inspects which best maximizes

$$P(k) * P(e | k)$$

The value of P(k) was obtained from Language Modeling and P(e|k) was obtained from Translation Modeling. But during decoding many values are obtained for different phrasal combination of Target language. The decoder selects the maximized value among them.

Let, P(k1)=0.13 and P(k2)=0.38 where 'k1' and 'k2' are the target sentences generated with different combination of kannada phrases. Let P(e| k1) is 0.71 and P(e | k2) is 0.49. 'e' is the given English sentence which doesn't change throughout the process, hence it remains constant.

Case 1: P(k1) * P(e|k1)=0.13*0.71=0.0923
Case 2: P(k2) * P(e|k2)=0.38*0.49=0.1862

As the maximum value is obtained for the Case 2 that is when Kannada sentence 'k2' is considered. The decoder outputs the 'k2' as the Best translated Target sentence.

8. Evaluation

As a primitive step to achieve SMT, we have taken a small sized corpus to get machine translation of simple sentences. Due to this the Evaluation with respect to BLEU score index becomes zero. In spite of the BLEU score index as zero the manual evaluation gets, the expected translation between source and target languages. Since we used a predefined set of sentences in our corpus. We can scale our system by increasing the corpus to get random translation between the sentences and then we can apply BLEU score metric to evaluate the system that will be considered in the later stage of study.

The BLEU metric measures how many words overlap in a given translation when compared to a reference translation, giving higher scores to sequential words.

The BLEU score for present experiment is zero. The reason behind the poor quality of corpus is explained below. Using a sentence-level version of BLEU is problematic though it can easily become zero. This is because of the product of n-gram precisions in the geometric mean of the precision component of BLEU: if any one parameter is zero, the whole product will be zero. In particular, it is easy to see that BLEU will be zero for any hypothesis without 4-gram matches. This is undesirable for optimization purposes since it does not allow distinguishing a hypothesis translation that has no matches at all and one that has unigram, bigram and trigram matches but no 4-gram matches [6].

There are many strategies to get non-zero BLEU score, like smoothing the BLEU or simply increasing the size of corpus etc. But these discussions fall outside the scope of the current study

9. Conclusion and future work

In this primitive work towards SMT on English to Kannada translation, we have succeeded to get the machine translation for a predefined set of sentences ranging from 3 to 8 words per sentence. The study gives a brief description of statistical aspects of machine translation. The BLEU scores obtained during the process of evaluation motivates for the increase in corpora size. By enhancing the size of the corpus we can get random and more accurate translation from English to Kannada language. Better aligned corpora and further experimentation may yield even better results in BLEU scores and more accuracy in translation. A good machine translation can be obtained by creating a domain specific well aligned corpora.

10. Acknowledgment

This paper is a continuation of the major project carried out at the Amrita Vishwa Vidya Peetham, Mysore. My special thanks are extended to Br. Sunil Dharmapal, Director of ASAS,Mysore, Br.Venugopal, Correspondent of ASAS,Mysore, Prof. Vidya Pai.C, Principal of ASAS Mysore and Mrs. Kanchana V, vice chairperson of Computer science department. Advice given by Mr. Shiva Kumar K M, Asst. Professor, Department of Computer Science has been a great help in successful of this project and paper.

References

[1] www.statmt.org/moses/
[2] English To Hindi Statistical Machine Translation System by Nakul Sharma
[3] Co-training for Statistical Machine Translation by Chris Callison-Burch
[4] The Mathematics of Statistical Machine Translation: Parameter Estimation by Peter E Brown, Stephen A. Della Pietra, Vincent J. Della Pietra and Robert L. Mercer.
[5] Statistical Machine Translation by Philip Koehn
[6] Optimizing for Sentence-Level BLEU+1 Yields Short Translations by Preslav Nakov, Francisco Guzma´N And Stephan Vogel
[7] A Statistical MT Tutorial Workbook by Kevin Knight
[8] Christopher D Manning and Hinrich Schutze, "Foundations of Statistical Natural Language processing", MIT Press, 1999.
[9] Daniel and James H Martin "speech and Language Processing: An Introduction to Natural Language processing, computational Linguistics and speech Recognition", second edition, Prentice Hall of India, 2008.

Author Profile

Shivakumar KM is pursuing his doctoral degree in computer science from Amrita Vishwavidyapeetham (University) Bangalore campus. He received his MCA from Visvesvaraya Technological University and M.phil from Madurai Kamaraj University in 2003 and 2007 respectively. Since 11 years he is teaching post graduate students like MCA and M.Sc. now he is with Amrita Vishwavidyapeetham.

Chitra C is an MCA student at Amrita Vishwavidyapeetham Mysore campus. She received her MCA from Amrita Vishwavidyapeetham in 2014.